Wordcloud

Joost Bloos

04/11/2021

```
#https://cran.r-project.org/web/packages/lexicon/index.html
#install.packages("lexicon", dependencies = TRUE)
#https://cran.r-project.org/web/packages/tm/index.html
#install.packages("tm", dependencies = TRUE)
#https://cran.r-project.org/web/packages/RWeka/index.html
#install.packages("RWeka", dependencies = TRUE)
#https://cran.r-project.org/web/packages/textstem/index.html
#install.packages("textstem", dependencies = TRUE)
#https://cran.r-project.org/web/packages/textclean/index.html
#install.packages("textclean", dependencies = TRUE)

#install.packages("dplyr")
#install.packages("quanteda")
#install.packages("textstem")
#install.packages("text2vec")
#install.packages("namespace")
#install.packages("stopwords")

#Loading the packages to the current workspace
lstPackages <- c('lexicon','tm','RWeka','textstem','textclean')

lapply(lstPackages, library, character.only = TRUE)
```

```
## Loading required package: NLP
```

```
## Loading required package: koRpus.lang.en
```

```
## Loading required package: koRpus
```

```
## Loading required package: sylly
```

```
## For information on available language packages for 'koRpus', run
##
##   available.koRpus.lang()
##
## and see ?install.koRpus.lang()
```

```
##
## Attaching package: 'koRpus'
## The following object is masked from 'package:tm':
##
##     readTagged

## [[1]]
## [1] "lexicon"   "stats"     "graphics"  "grDevices" "utils"     "datasets"
## [7] "methods"   "base"
##
## [[2]]
##  [1] "tm"        "NLP"       "lexicon"   "stats"     "graphics"  "grDevices"
##  [7] "utils"     "datasets"  "methods"   "base"
##
## [[3]]
##  [1] "RWeka"     "tm"        "NLP"       "lexicon"   "stats"     "graphics"
##  [7] "grDevices" "utils"     "datasets"  "methods"   "base"
##
## [[4]]
##  [1] "textstem"       "koRpus.lang.en" "koRpus"         "sylly"
##  [5] "RWeka"          "tm"             "NLP"            "lexicon"
##  [9] "stats"          "graphics"       "grDevices"      "utils"
## [13] "datasets"       "methods"        "base"
##
## [[5]]
##  [1] "textclean"      "textstem"       "koRpus.lang.en" "koRpus"
##  [5] "sylly"          "RWeka"          "tm"             "NLP"
##  [9] "lexicon"        "stats"          "graphics"       "grDevices"
## [13] "utils"          "datasets"       "methods"        "base"
```

```
library(quanteda)
```

```
## Package version: 3.1.0
## Unicode version: 13.0
## ICU version: 69.1

## Parallel computing: 4 of 4 threads used.

## See https://quanteda.io for tutorials and examples.

##
## Attaching package: 'quanteda'

## The following objects are masked from 'package:koRpus':
##
##     tokens, types

## The following object is masked from 'package:tm':
##
##     stopwords

## The following objects are masked from 'package:NLP':
##
##     meta, meta<-
```

```
library(stringr)
library(dplyr)
```

```
##
## Attaching package: ’dplyr’

## The following objects are masked from ’package:stats’:
##
##     filter, lag

## The following objects are masked from ’package:base’:
##
##     intersect, setdiff, setequal, union
```

```
library(wordcloud)
```

```
## Loading required package: RColorBrewer
```

```
library(textstem)

#For successful knitting of document in pdf:
#tinytex::install_tinytex()
```

```
#read data set Tweets May 16, 2020: Covid related hastags as per project document.

data_set_may <- read.csv("corona_tweets_59 May 2020", header = T, sep = ",")
```

```
## Warning in scan(file = file, what = what, sep = sep, quote = quote, dec = dec, :
## embedded nul(s) found in input
```

```
#take a sample of 1,000, set seed to replicate results across several analysis of methods:
set.seed(1000)
data_may <- data_set_may[sample(nrow(data_set_may), size = 1000), ]
str(data_may)
```

```
## ’data.frame':    1000 obs. of  35 variables:
##  $ coordinates              : chr  "" "" "" "" ...
##  $ created_at               : chr  "Sat May 16 23:31:16 +0000 2020" "Sat May 16 18:57:19 +0000 2020
##  $ hashtags                 : chr  "" "" "" "" ...
##  $ media                    : chr  "" "" "" "" ...
##  $ urls                     : chr  "" "" "" "https://www.nbcnews.com/now/video/officials-warn-chine
##  $ favorite_count           : int  0 0 0 0 0 0 0 0 1 1 ...
##  $ id                       : num  1.26e+18 1.26e+18 1.26e+18 1.26e+18 1.26e+18 ...
##  $ in_reply_to_screen_name  : chr  "" "" "" "" ...
##  $ in_reply_to_status_id    : num  NA NA NA NA NA NA NA NA NA NA ...
##  $ in_reply_to_user_id      : num  NA NA NA NA NA NA NA NA NA NA ...
##  $ lang                     : chr  "en" "en" "en" "en" ...
##  $ place                    : chr  "" "" "" "" ...
##  $ possibly_sensitive       : chr  "" "" "" "false" ...
##  $ quote_id                 : num  NA NA NA NA 1.26e+18 ...
## $ retweet_count           : int  25 338 441 0 0 12022 4 11 1 0 ...
## $ retweet_id              : num  1.26e+18 1.26e+18 1.26e+18 NA NA ...
## $ retweet_screen_name     : chr  "business" "Suewilson91" "BreitbartNews" "" ...
## $ source                  : chr  "<a href=\"http://twitter.com/download/iphone\" rel=\"nofollow\">
## $ text                    : chr  "Many Americans have proven diligent in staying home to limit the
## $ tweet_url               : chr  "https://twitter.com/lemnosalt/status/1261801422430978048" "https
## $ user_created_at         : chr  "Tue Feb 10 00:25:20 +0000 2009" "Sun Dec 01 15:12:16 +0000 2019
## $ user_id                 : num  2.05e+07 1.20e+18 8.17e+17 1.66e+09 1.26e+18 ...
## $ user_default_profile_image: chr  "false" "false" "false" "false" ...
## $ user_description        : chr  "Groovy chick and media producer. All snark. No bite." "" "" "Jus
## $ user_favourites_count   : int  92045 19675 1 46635 2788 1371 1230 18960 4 34505 ...
## $ user_followers_count    : int  1469 45 65 263 426 97 109 2151 375 12607 ...
## $ user_friends_count      : int  2526 229 228 1960 267 240 274 4846 227 12722 ...
## $ user_listed_count       : int  73 0 1 1 4 1 0 15 13 106 ...
## $ user_location           : chr  "" "New Forest" "" "United States" ...
## $ user_name               : chr  "Lynn" "Hilary ðŸ’\231" "Bill Spears" "Bet" ...
## $ user_screen_name        : chr  "lemnosalt" "Hilary72926522" "BillSpears724" "Bet_the_ChE" ...
## $ user_statuses_count     : int  35678 5272 24796 23697 1028 317 279 84594 14606 252203 ...
## $ user_time_zone          : logi  NA NA NA NA NA NA ...
## $ user_urls               : chr  "http://lynnmargherita.com" "" "" "" ...
## $ user_verified           : chr  "false" "false" "false" "false" ...
```

```
#Add column index to transform file to format appropriate for corpus
data_may$index <- 1:nrow(data_may)
str(data_may)
```

```
## 'data.frame':    1000 obs. of  36 variables:
## $ coordinates             : chr  "" "" "" "" ...
## $ created_at              : chr  "Sat May 16 23:31:16 +0000 2020" "Sat May 16 18:57:19 +0000 2020
## $ hashtags                : chr  "" "" "" "" ...
## $ media                   : chr  "" "" "" "" ...
## $ urls                    : chr  "" "" "" "https://www.nbcnews.com/now/video/officials-warn-chines
## $ favorite_count          : int  0 0 0 0 0 0 0 0 1 1 ...
## $ id                      : num  1.26e+18 1.26e+18 1.26e+18 1.26e+18 1.26e+18 ...
## $ in_reply_to_screen_name : chr  "" "" "" "" ...
## $ in_reply_to_status_id   : num  NA NA NA NA NA NA NA NA NA NA ...
## $ in_reply_to_user_id     : num  NA NA NA NA NA NA NA NA NA NA ...
## $ lang                    : chr  "en" "en" "en" "en" ...
## $ place                   : chr  "" "" "" "" ...
## $ possibly_sensitive      : chr  "" "" "" "false" ...
## $ quote_id                : num  NA NA NA NA 1.26e+18 ...
## $ retweet_count           : int  25 338 441 0 0 12022 4 11 1 0 ...
## $ retweet_id              : num  1.26e+18 1.26e+18 1.26e+18 NA NA ...
## $ retweet_screen_name     : chr  "business" "Suewilson91" "BreitbartNews" "" ...
## $ source                  : chr  "<a href=\"http://twitter.com/download/iphone\" rel=\"nofollow\">
## $ text                    : chr  "Many Americans have proven diligent in staying home to limit the
## $ tweet_url               : chr  "https://twitter.com/lemnosalt/status/1261801422430978048" "https
## $ user_created_at         : chr  "Tue Feb 10 00:25:20 +0000 2009" "Sun Dec 01 15:12:16 +0000 2019
## $ user_id                 : num  2.05e+07 1.20e+18 8.17e+17 1.66e+09 1.26e+18 ...
## $ user_default_profile_image: chr  "false" "false" "false" "false" ...
## $ user_description        : chr  "Groovy chick and media producer. All snark. No bite." "" "" "Jus
## $ user_favourites_count   : int  92045 19675 1 46635 2788 1371 1230 18960 4 34505 ...
## $ user_followers_count    : int  1469 45 65 263 426 97 109 2151 375 12607 ...
## $ user_friends_count      : int  2526 229 228 1960 267 240 274 4846 227 12722 ...
##  $ user_listed_count        : int  73 0 1 1 4 1 0 15 13 106 ...
##  $ user_location            : chr  "" "New Forest" "" "United States" ...
##  $ user_name                : chr  "Lynn" "Hilary ðŸ’\231" "Bill Spears" "Bet" ...
##  $ user_screen_name         : chr  "lemnosalt" "Hilary72926522" "BillSpears724" "Bet_the_ChE" ...
##  $ user_statuses_count      : int  35678 5272 24796 23697 1028 317 279 84594 14606 252203 ...
##  $ user_time_zone           : logi  NA NA NA NA NA NA ...
##  $ user_urls                : chr  "http://lynnmargherita.com" "" "" "" ...
##  $ user_verified            : chr  "false" "false" "false" "false" ...
##  $ index                    : int  1 2 3 4 5 6 7 8 9 10 ...
```

```
#set of Corpus using VectorSource() and VCorpus
listofDocs <- tm::VectorSource(data_may$text)
listofDocs$Names <- names(data_may$index)
corporaData <- tm::VCorpus(listofDocs)
#use VCorpus as it allows for customized tokenization required for n-gram analysis later on in the code
```

```
#Lemmatization is the process of reducing a word to its base form while incorporating information about
#Utilizing Thesaurus: lexicon
for(i in 1:1000)
{
    corporaData[[i]]$content <-
    textstem::lemmatize_strings(corporaData[[i]]$content,
                                dictionary = lexicon::hash_lemmas)
}

#Stemming removes a word's suffix (ending), such as es, s, ing, ed, y, based on an heuristic algorithm.

corporaData <- tm::tm_map(corporaData, stemDocument)

#COULDN'T RESOLVE KNITTING ERROR: error in match.fun(FUN) : object 'stemdocument' not found

#remove words that don't add to context of Tweet, but more so are terms that don't distinguish well bet
#Stopword Removal

corporaData <- tm::tm_map(corporaData, removeWords, stopwords('english'))

#Other Pre-processing Steps: Punctuation Marks, Extra Whitespaces, etc
corporaData <- tm::tm_map(corporaData, content_transformer(tolower))
corporaData <- tm::tm_map(corporaData, removePunctuation,
                      ucp = TRUE,
                      preserve_intra_word_contractions = FALSE,
                      preserve_intra_word_dashes = FALSE)
corporaData <- tm::tm_map(corporaData, removeNumbers)
corporaData <- tm::tm_map(corporaData, stripWhitespace)

#moving to end as it created better results:
corporaData <- tm::tm_map(corporaData, removeWords, stopwords('SMART')) #error: source not found, wasn'
```

```
## Warning: 'stopwords(language = "SMART")' is deprecated.
## Use 'stopwords(source = "smart")' instead.
## See help("Deprecated")
```

```r
corporaData[[1]]$content
```

```
## [1] "mani american prove dilig stay home limit spread covid  accept social distanc  wane https   diq
```

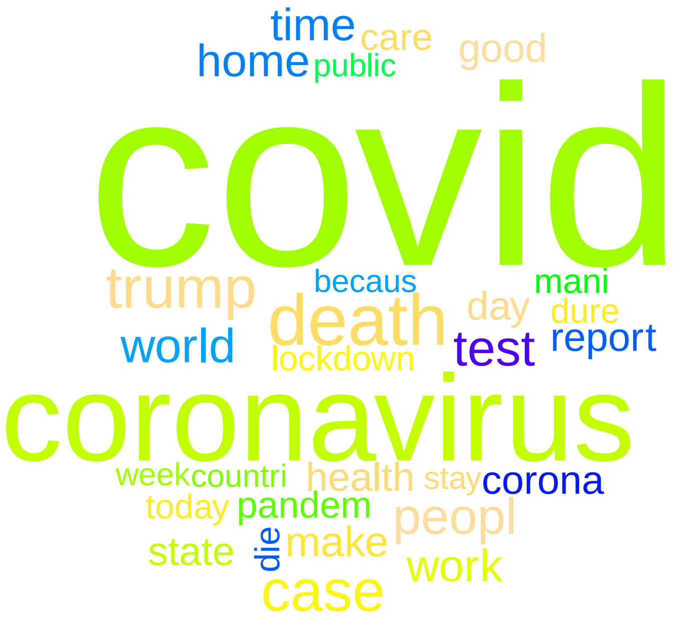
```r
#data preprocessing or text normalization:
#Social media text may need additional cleansing to remove links, hashtags, retweets, social media hand

#Creating another corpus reference to be used for wordcloud

tweets_corpus_may <-corporaData
```

```r
#I wanted to do other clean-up of terms like "http" and "amp", but kept getting error: Error in UseMeth

#to remove other characters as per output wordcloud:
toSpace <- function(x, pattern) gsub(pattern, " ", x)
tweets_corpus_may <- tm_map(tweets_corpus_may, toSpace, "ÿ")
tweets_corpus_may <- tm_map(tweets_corpus_may, toSpace, "amp")
tweets_corpus_may <- tm_map(tweets_corpus_may, toSpace, """)
tweets_corpus_may <- tm_map(tweets_corpus_may, toSpace, "Itâ€")
tweets_corpus_may <- tm_map(tweets_corpus_may, toSpace, "â")
```

```r
#I tried running the code above and below but getting the following error: Error in UseMethod("inspect"
#After research, it appears that the code is rewriting the object to another data type. Then I used cor

# remove retweets
tweets_corpus_may <- tm_map(tweets_corpus_may, (function(x) gsub('\\b+RT', " ", x)))
# remove mentions
tweets_corpus_may <- tm_map(tweets_corpus_may, (function(x) gsub('@\\S+', " ", x)))
# remove hashtags
tweets_corpus_may <- tm_map(tweets_corpus_may, (function(x) gsub('#\\S+', " ", x)))

# remove links
tweets_corpus_may <- tm_map(tweets_corpus_may, (function(x) gsub("http[^[:space:]]*", " ", x))) #double

#For sentiment analysis only: https://rpubs.com/chelseyhill/669117 not all preprocessing steps appropri
```

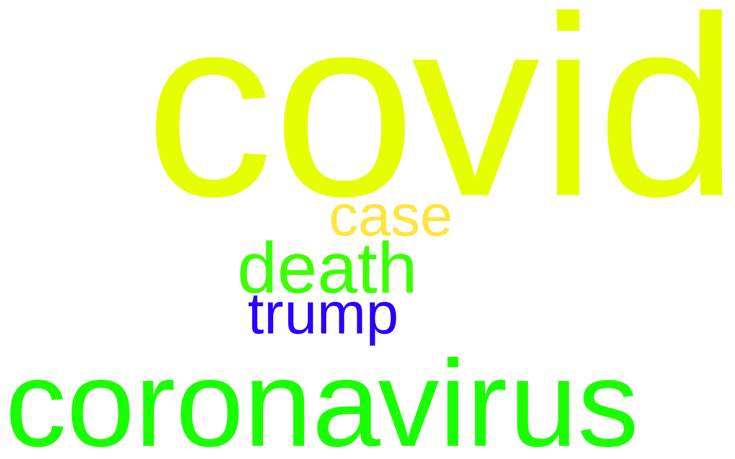
```r
#to correct error message to apply correct data type after function "tolower"
tweets_corpus_may_worldcloud <- tm_map(tweets_corpus_may, PlainTextDocument)
```

```r
#Various world clouds min and max term frequency adjusted:
wordcloud(tweets_corpus_may_worldcloud, max.words = 30, scale = c(8, .5), colors = topo.colors(n=30), ra
```



```
# There are two words in particular "https" and "amp" that i was able to remove while fine tune the pre
```

```
wordcloud(tweets_corpus_may_worldcloud, min.freq = 75, max.words = 30, scale = c(8, 0.5), colors = topo
```


covid
case
death
trump
coronavirus